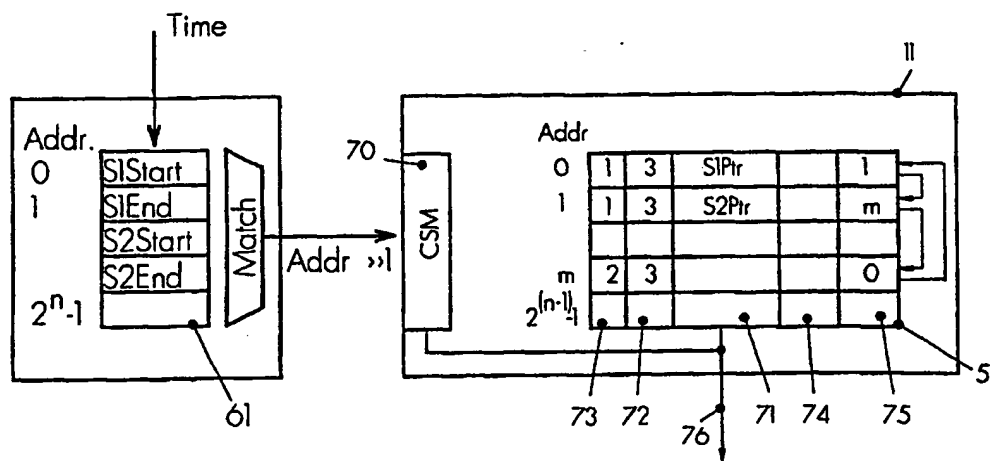




INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(51) International Patent Classification ⁶ : G06F 17/30	A1	(11) International Publication Number: WO 95/25313 (43) International Publication Date: 21 September 1995 (21.09.95)
(21) International Application Number: PCT/EP94/00833 (22) International Filing Date: 16 March 1994 (16.03.94) (71) Applicant (for all designated States except US): INTERNATIONAL BUSINESS MACHINES CORPORATION [US/US]; Old Orchard Road, Armonk, NY 10504 (US). (72) Inventors; and (75) Inventors/Applicants (for US only): KAISERSWERTH, Matthias [DE/CH]; Leigrubenstrasse 2, CH-8805 Richterswil (CH). RÜTSCHÉ, Erich [CH/FR]; 1500, chemin des Adrets, F-06370 Mouans-Sartoux (FR). (74) Agent: BARTH, Carl, O.; IBM Research Laboratory, Intellectual Property Dept., Säumerstrasse 4, CH-8803 Rüschlikon (CH).		(81) Designated States: JP, US, European patent (AT, BE, CH, DE, DK, ES, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE). Published With international search report.

(54) Title: METHOD AND APPARATUS FOR SYNCHRONIZATION AND SCHEDULING OF MULTIPLE DATA STREAMS AND REAL TIME TASKS



(57) Abstract

The present application concerns a method and an apparatus for the synchronization and the scheduling of multiple data streams and real time tasks. In accordance with the present invention, pointers pointing to the information streams to be synchronized and processed are kept in the table (51) of a selector (11). The time marks (SIStart etc.) of said information streams are stored in a table (61) of a timer. The system time is checked against said time marks and if a time mark has been reached, the table (51) of said selector (11) is updated. If the status of an information stream is such that its trigger condition is fulfilled, the pointer to said information stream is made available at the output of the selector (11). This pointer is then used to retrieve the respective information stream and to forward it to a payout device.

FOR THE PURPOSES OF INFORMATION ONLY

Codes used to identify States party to the PCT on the front pages of pamphlets publishing international applications under the PCT.

AT	Austria	GB	United Kingdom	MR	Mauritania
AU	Australia	GE	Georgia	MW	Malawi
BB	Barbados	GN	Guinea	NE	Niger
BE	Belgium	GR	Greece	NL	Netherlands
BF	Burkina Faso	HU	Hungary	NO	Norway
BG	Bulgaria	IE	Ireland	NZ	New Zealand
BJ	Benin	IT	Italy	PL	Poland
BR	Brazil	JP	Japan	PT	Portugal
BY	Belarus	KE	Kenya	RO	Romania
CA	Canada	KG	Kyrgyzstan	RU	Russian Federation
CF	Central African Republic	KP	Democratic People's Republic of Korea	SD	Sudan
CG	Congo	KR	Republic of Korea	SE	Sweden
CH	Switzerland	KZ	Kazakhstan	SI	Slovenia
CI	Côte d'Ivoire	LI	Liechtenstein	SK	Slovakia
CM	Cameroon	LK	Sri Lanka	SN	Senegal
CN	China	LU	Luxembourg	TD	Chad
CS	Czechoslovakia	LV	Latvia	TG	Togo
CZ	Czech Republic	MC	Monaco	TJ	Tajikistan
DE	Germany	MD	Republic of Moldova	TT	Trinidad and Tobago
DK	Denmark	MG	Madagascar	UA	Ukraine
ES	Spain	ML	Mali	US	United States of America
FI	Finland	MN	Mongolia	UZ	Uzbekistan
FR	France			VN	Viet Nam
GA	Gabon				

1

DESCRIPTION

5

**Method and Apparatus for Synchronization and Scheduling
of Multiple Data Streams and Real Time Tasks****TECHNICAL FIELD**

10 The present invention concerns a method and apparatus for the synchronization and the scheduling of multiple data streams as well as for the scheduling of tasks in operating systems with hard real-time requirements.

15

BACKGROUND OF THE INVENTION

Multimedia information is composed of several objects that constitute a document. These objects could be of different data types, such as text, still images, video, and sound. In order for these objects to be meaningful to a user, there must be a logical and temporal structure of the objects constituting the documents' contents. To ensure this logical and temporal structure, e.g. when displaying a video sequence on a display device and providing a soundtrack for this video sequence via an adapter card to a speaker, the respective data streams have to be synchronized and scheduled before being sent to the output devices. The same principle applies to other related multimedia information which has to be synchronized in communicating with other devices or users (e.g. teleconferencing), before processing and presentation.

30 Synchronization and scheduling is a critical task for a wide spectrum of applications ranging from computer-integrated telephony, text with voice annotations, any kind of multi media systems, and interactive services (e.g. interactive television) to cooperative teleworking on documents including

1 foils and video, and distributed games. A further application, where
information streams have to be synchronized, is known as medical imaging,
for which high-resolution images must be distributed reliably and fast
throughout a hospital.

5

To date, the synchronization is typically performed in software. However, if
the scheduling intervals are too short, or if too many information streams
must be synchronized, a software solution, which is very CPU intensive, is
often the limiting factor as far as processing speed is concerned.

10

In real-time systems time-critical tasks have to be scheduled reliably and
fast. In the present context, any system in which the time at which an output
is produced is significant, is referred to as real-time system. The lag from
the input time of a real-time system to its output time, i.e. the delay, must be
15 sufficiently small for acceptable timeliness. With other words, a real-time
system has to respond to an externally-generated input stimuli within a
finite and specified period. In hard real-time systems, i.e. in those systems
where it is absolutely imperative that responses occur within a specified
deadline, scheduling of time-critical tasks is of particular interest. However,
20 the present invention is not limited to hard real-time systems. It might be
used in soft real-time systems, too. Such a soft real-time system would still
function correctly if deadlines are occasionally missed, and a hardware
solution in accordance with the present invention might be to costly under
certain circumstances.

25

SUMMARY OF THE INVENTION

It is an object of the present invention to provide a method and apparatus
30 for off-loading the synchronization of multiple information streams from the
processor of the system handling these information streams.

1 It is an object of the present invention to provide a method and apparatus
for fast and reliable synchronization and scheduling of multiple information
streams.

5 It is another object of the present invention to provide a method and
apparatus for scheduling tasks or processes in real-time systems.

The above objects have been accomplished by handling pointers pointing to
the information streams to be synchronized and by maintaining a table with
10 status information of each information stream. The time marks of said
information streams are stored in a table and the system time is checked
against said time marks. If a time mark has been reached the status
information of the respective information stream is updated. If the status of
an information stream is such that its trigger condition is fulfilled, the
15 pointer to said information stream is made available for playout of this
information stream.

20

25

30

1

DESCRIPTION OF THE DRAWINGS
AND NOTATIONS USED

The invention is described in detail below with reference to the following
5 drawings:

- FIG. 1** shows a system, in accordance with the present invention, for
payout of two information streams.
- 10 **FIG. 2** illustrates the synchronization of multiple data streams.
- FIG. 3** illustrates the synchronization and play out of two data streams
when the respective trigger condition is fulfilled.
- 15 **FIG. 4** illustrates how a priority value might be increased with
advancing time.
- FIG. 5** shows the two core components of an apparatus for
synchronization and scheduling of data streams and real time
20 tasks, in accordance with the present invention.
- FIG. 6A** shows a first exemplary embodiment of the timer.
- FIG. 6B** shows a second exemplary embodiment of the timer.
- 25 **FIG. 7** shows an embodiment of the timer and selector, in accordance
with the present invention.
- FIG. 8** shows an example of an 8-bit timer where every second bit is
30 used.

GENERAL DESCRIPTION

In multimedia systems multiple data streams must be synchronized and scheduled for playout to, for example, a speaker 15 and a video display 14, as illustrated in Figure 1. The synchronization of multiple data streams is illustrated in Figure 2. The three streams are synchronized at so-called synchronization points (SyncPoint) 22. A stream may be synchronized to one or multiple other streams or to time stamps 23 defined by an external clock. This synchronization is typically performed in software, as mentioned in the introductory portion of the present description. In the following, a mechanism is presented which performs synchronization and output scheduling of multiple data streams in hardware without central processing unit (CPU) interaction. The same mechanism can also be used for scheduling tasks in real-time systems.

A data stream is a sequence of data units. In Figure 3, the synchronization of two data units is illustrated. The SyncPoints are translated to time marks relative to the system which processes the received data streams. The SyncPoints might be translated to time marks by means of a clock or counter of said system. In the example given in Figure 3, a data unit from stream 1 cannot be played out before a time mark 30 (S1Start) and it must be played out before a time mark 31 (S1End). The same applies to the second data stream (stream 2). As illustrated in Figure 3, at least two different states (WAIT, READY) can be associated with a data stream. Before SxStart of a data stream x has been reached, the respective data unit is in the WAIT state. After SxStart and before SxEnd it is in the READY state. It is conceivable to assign further states, such as a DELAYED state, to these data streams. The DELAYED state might be associated to a data unit after SxEnd.

In order to be able to synchronize two data units for playout, trigger conditions are defined in accordance with the present invention. A typical trigger condition might be:

1

IF both data units are in the READY state
THEN they can be played out.

5

The trigger conditions are checked and as soon as the condition is fulfilled, the data units are played out. This concept is schematically illustrated in Figure 3, where means for checking the trigger condition 32 are shown.

10

In general, two or more data units can be related in time by trigger conditions, which can be an arbitrary logical combination of the different states of the respective data units. Fulfilling a given trigger condition then initiates the appropriate operation on the units.

15

The synchronization can be further enhanced with a priority value P to distinguish between data units (or tasks) which fulfill a trigger condition to select an older or more important stream (or task). This, for example, allows to prioritize streams that should not be delayed, whereas other data streams might be delayed. In Figure 4 an example is shown how the priority value P of a data unit (or task) can be increased in time. In this example, three different states (WAIT, READY, DELAYED) are associated with the data stream. The priority value P might also be increased or decreased by interaction of a processor or upon receipt of a feedback signal of a playout device.

25

A first embodiment of the present invention is schematically illustrated in Figure 1. According to this example, two data streams are received by means 18 for extraction of time marks. The first data stream originates from a storage disk 16, whereas the second data stream is sent via a network 17 to said means 18 for extraction of time marks. After extraction of the time marks, the data units of the two data streams are stored in buffers 19 and 20, and the respective time marks are provided to a timer 10. The buffer addresses are forwarded to a selector 11. In addition, a system clock or counter 13 is connected to said timer 10, and a processor, e.g. a CPU, might

30

1 be linked to the timer 10 and selector 11. Two playout devices, a display unit 14 and a loudspeaker 15, are connected via an interfacial playout device to the buffers 19 and 20.

5 The two core components of the present invention are the timer 10 and the selector 11. Their operation is described in connection with Figure 5.

The timer 10 holds a table 50 with entries concerning the time marks of each data unit received and stored in a buffer. These time marks are relative to a time signal either being generated by a system clock 13 or a counter. Each time mark reached is then output to the selector 11. The table 50 is written by an external processor. The selector comprises a table 51 where for each data unit in a buffer a pointer to said data unit, i.e. the respective buffer address, the state of said data unit, and the trigger condition are held. Data units to be synchronized are linked into a circular list via their table entries. For each time mark received from the timer 10, the sector 11 updates the status field of the concerned data unit and then checks the trigger condition of all data units to be synchronized with said data unit. Once the trigger condition is fulfilled, the pointer to the data unit (or units) is made available at the output of selector 11. The output of the pointer then triggers retrieval and playout of the data unit by means of the playout device 21. Table 51 of the selector 11 might be maintained and updated by an external processor, e.g. CPU 12.

25 The timer 10 reads the current system time and returns the address of a unit pointer in the selector 11. This function can be provided in several ways. First, a solution based on a content addressable memory (CAM) is presented which is suitable for a high resolution timer but relatively expensive. The second solution is less expensive and is suitable for lower resolution timers and for implementing timer priority as described below. The two solutions are shown in Figures 6A and 6B, respectively.

1 The CAM timer 60 is illustrated in Figure 6A. It comprises a CAM 61. The
start and end time mark for each data unit to be scheduled are written into
consecutive addresses of the CAM 61. The addresses then differ only in the
least significant bit. Thus, for a CAM of 2^n addresses, $n-1$ bits determine the
5 address of the scheduling unit pointer in the selector table 51. The width of
a CAM entry is determined by the resolution of the time signal received at
input 62. For a 32-bit time signal also a 32-bit wide CAM 61 is required.

For lower time resolution a timer 63 with counter 64, as shown in Figure 6B,
10 might be employed. In this case the difference between the current time
and the time mark is calculated and its two's complement is written to the
timer table 65. The counter 64 increments each entry for each time tick at
input 66. If the carry bit of an entry becomes set, its address is shifted right
by one bit and output to the selector 11. As the counter 64 continues to
15 increment the entries, the content of the entry (with its carry bit = 1) can
then be used to reflect the time passed since reaching the start and/or end
time. The counter value thus describes a time based priority. With this
counter value a priority value P might be generated. The width of a timer
entry can range from 8 to 32 bits depending on the timer resolution and the
20 required timer differences. A free flag 67 might be provided in order to
mark free table entries in table 65. This flag 67 is used by the counter to
determine which entries it must increment.

For both schemes, a FIFO can hold free timer table entries to reduce the
25 management load of the external processor. The processor gets a free entry
from the FIFO and writes delayed entries, or entries which are freed, to the
FIFO. The selector might directly write delayed entries to this FIFO.

It is obvious that the present invention is not limited to the above two
30 implementations of timers since other timers are conceivable which are well
suited, too.

1 The selector 11 comprises a table 51 and a comparator state machine (CSM)
 70. The table 51 has $2^{(n-1)}$ entries (rows or columns). The exemplary
 implementation of the selector 11, given in Figure 7, shows a table 51 with
 $2^{(n-1)}$ rows. Each row has a pointer field 71 holding the pointer, i.e. the
 5 buffer address S1Ptr, to the data unit to be scheduled, a p-bit wide priority
 field 72, a k-bit wide status field 73, a 2-bit wide trigger field 74, and an
 (n-1)-bit wide synchronization field 75. The width of said status field 73
 depends on the number of different states. In case of three different state
 (WAIT, READY, DELAYED) the status field 73 is 2-bits wide. The
 10 synchronization field 75 holds a pointer linking all those entries with which a
 data unit must be synchronized into a circular list. This circular list is
 schematically illustrated by means of arrows on the right hand side of table
 51. The CSM 70 receives the address of a unit pointer from the timer 10
 and updates the state of the status field 73 in this row. The CSM 70 then
 15 follows the linked list of synchronizations entries and computes whether the
 trigger condition is fulfilled by comparing the status fields 73 with the trigger
 fields 74. The trigger condition might also be checked by means of simple
 logical elements such as AND-gates, OR-gates etc.

20 In the following, it is assumed, without loss of generality, that the trigger
 condition is fulfilled when all entries in the respective trigger fields 74 are in
 the READY state. In this case, the trigger state fields are redundant and
 might be omitted.

25 Once all the entries linked by the synchronization field 75 are READY, the
 CSM 70 outputs the entries according to their priorities, if any. With $p=4$, 16
 priority classes can be distinguished. The width w of the scheduling unit
 pointer is determined by the memory (buffer) architecture of the system. The
 width s of the synchronization field 75 is determined by the size of the
 30 selector table 51, with $s=n-1$.

In Figure 7 the problem shown in 3 is elaborated. The time marks 30
 (S1Start) and 31 (S1End) of data unit S1 are written to the addresses 0 and 1

1 of the CAM 61. The address of the pointer to the data unit S1 is the CAM
address shifted right by one bit. An external processor 12 writes the time
marks to the timer table, i.e. the CAM 61 and the entries to the selector
table 51. For a data unit to be scheduled which requires priority, the priority
5 field 72 is set. In this example a 4-bit wide priority field 72 is used. For each
time mark reached, the state, reflected by the status field 73, is updated. If
the state is READY, the CSM 70 reads the synchronization field 75 and
follows the linked list to determine whether the state of all other entries
fulfills the trigger condition. If the trigger condition is reached, the unit
10 pointer with the highest priority is made available at output 76 first. If
multiple units have the same priority, or in case that no priority fields are
provided, the unit pointers might either be output in parallel or in a
consecutive manner, e.g. by order of their position in the selector table 51. It
is obvious that the priority fields 72 can be omitted, too.

15 In many systems, multimedia as well as real-time, the scheduling of actions,
data streams, tasks, processes, or the playout of data units is determined by
a priority which increases with time. If a counter timer, as illustrated in
Figure 6B, is used as the timer component 10, the entry in the timer table 65
20 also describes its priority. This priority can be used in the priority field 72 of
the selector 11 to achieve time dependent scheduling. The four most
significant bits of the timer entry are used to distinguish again 16 priorities.
A different set of bits might be used if the priority must increase
non-linearly. Figure 8 shows an example of an 8-bit timer where every
25 second bit is used to get a priority mechanism that maps the 256 timer
values to 16 non-linear priority values P.

30 According to the present invention, a synchronization point (SyncPoint) is
set up by writing the time marks to the free entries in the timer table and
setting up the associated entries in the selector table. For each stream this
takes one write operation to the timer and to the selector table, plus the
operations to set up the selector table entry. The actual scheduling takes
place in hardware without any further intervention of an external processor.

1 For a solution in software a similar number of setup operations would be
necessary but the search in a list of timers would take much longer.
Typically timers involve interrupt processing which is very time consuming
in particular in modern RISC processors with a large register file and
5 instruction pipeline. In the present invention these operations are avoided
and implemented in hardware. Additionally, the present invention can
handle timers with much finer granularity than any software solution.

The present invention, which has been described by means of a multimedia
10 related example, can be used in various multimedia communication
subsystems, e.g. for multimedia multiparty teleconferencing, or multimedia
protocol adapters. In ST-II and RTP protocol systems, for example, a playout
time stamp can be extracted from the protocol header and used in the
present scheduler (ST-II is the second version of an experimental Internet
15 stream protocol, and RTP is a transport protocol for real-time applications).
The present invention might also be used in multimedia bus systems which
provide a time signal. Data units to be triggered at a specified time are
written to the scheduler and the multimedia bus provides the timer input to
the timer unit. The invention can also be used in a multimedia adapter such
20 as the IBM ActionMedia II adapter (see IBM ActionMedia II Technical
Reference, IBM 04G5144, 2nd Edition, June 1992, Mechanicsburg), or the
MWave System (see "MWave Multimedia System", Technical Brief, Texas
Instruments, 1992) to synchronize multiple data streams. The term
ActionMedia is a trademark of the Intel Corporation and MWave is a
25 trademark of the IBM Corporation.

As already mentioned in the introduction, the present invention might also
be used in real-time systems for the scheduling of time-critical tasks or
processes. For these kind of application there is no synchronization field in
30 the table of the selector required. The CSM selects from the selector table
the entry with the highest priority. The scheduling unit pointer, i.e. a pointer
pointing to a task control block or to executable code, is then output. The
timer priority mechanism described above can be used to implement

- 1 scheduling policies with an explicit deadline, e.g. the earliest-deadline-first
or least-slack time scheduling algorithm as disclosed in the book "Real-Time
Systems and their Programming Languages", by A. Burns and A. Wellings,
Addison-Wesley Publishing Company, International Computer Science
5 Series, 1990. An external processor calculates the difference to the time
marks and writes it to the timer table. The entries are then used to derive
the priority of a job.

10

15

20

25

30

1 CLAIMS

- 1
1. Apparatus for synchronization of at least two information streams which comprises:
- 5 a) a timer (10) for holding a first table (50) with time marks for each one of said information streams, and for making each time mark reached available at the timer output,
- b) a selector (11) for holding a second table (51) with
- pointer fields (71) carrying pointers to each of said information streams,
 - 10 • status fields (73) carrying information concerning the status of each of said information streams, the status field of an information stream being updated each time a time mark of said information stream is reached and made available at said
- 15 timer output,
- c) means for checking whether a trigger condition of an information stream is fulfilled by watching the corresponding status field, and
- d) means for making the pointer to an information stream whose trigger condition is fulfilled available for playout of said information
- 20 stream.
2. The apparatus of claim 1, wherein said second table (50) further comprises:
- trigger fields (74) carrying information concerning the trigger
- 25 conditions of each of said information streams.
3. The apparatus of claims 1 or 2, wherein said second table (51) further comprises:
- synchronization fields (74) carrying, by means of a linked list,
- 30 entries concerning information streams to be played out in a certain relationship to each other.

- 1 4. The apparatus of claims 1, 2, or 3, wherein said second table (51)
further comprises:
- priority fields (72) carrying priority values (P) such that the
information stream with highest priority is played out first if the
5 respective trigger condition has been fulfilled.
5. The apparatus of claim 4, further comprising means for increasing said
priority values (P) with advancing time.
- 10 6. The apparatus of any of the preceding claims, further comprising:
- means (18) for extracting said time marks from incoming
information streams,
 - buffer means (19, 20) for storing said information streams,
 - playout means (21) for retrieval of said information streams stored
15 in said buffer means (19, 20) and for playout to the respective
playout devices (14, 15).
7. A multimedia system comprising an apparatus in accordance with any
of the preceding claims.
- 20 8. Apparatus for scheduling at least two real time tasks or two processes
which comprises:
- a) a timer (10) for holding a first table (50) with time marks for each
one of said tasks or processes, and for making each time mark
25 reached available at the timer output,
 - b) a selector (11) for holding a second table (51) with
 - pointer fields (71) carrying pointers to task control blocks or
executable code,
 - means for making a pointer to a task control block or an
30 executable code available at an output of said selector each
time a time mark of said task or process is reached in said
timer.

- 1 9. The apparatus of claim 8, wherein the pointer to a task or process with highest priority is made available first.
- 5 10. Method for synchronization of at least two information streams, characterized by the following steps:
- 10 a) checking whether a time mark of one of said information streams has been reached,
- 10 b) updating the status information of said information stream whose time mark has been reached,
- 10 c) determining whether the trigger condition of said information stream is fulfilled by watching said status information, and
- 10 d) providing a pointer to said information stream for further processing if the trigger condition of said information stream is fulfilled.
- 15 11. The method of claim 10, wherein the playout of information streams, which have to be played out in a predefined relationship to each other, is scheduled by means of a linked list being organized such that each time when a status information of one of said information streams is updated, the status information of the other information streams are
- 20 checked, and said information streams played out when all trigger conditions are fulfilled.
- 25 12. The method of claim 10, wherein a priority value (P) is assigned to each of said information streams, the information streams with highest priority being played out first.
- 30 13. The method of claim 12, wherein the priority value (P) of an information stream is increased with advancing time.
- 30 14. The method of claim 10, further comprising the following steps:
- extracting and/or synchronizing the time marks of said information streams,
 - storing said information streams in buffers,

- 1 • making the addresses of said buffers available as pointers for later
retrieval of said information streams,

5 15. The method of claim 14, wherein said information stream is retrieved
from said buffer by means of said pointer, prior to being send to a
playout device.

10

15

20

25

30

1/4

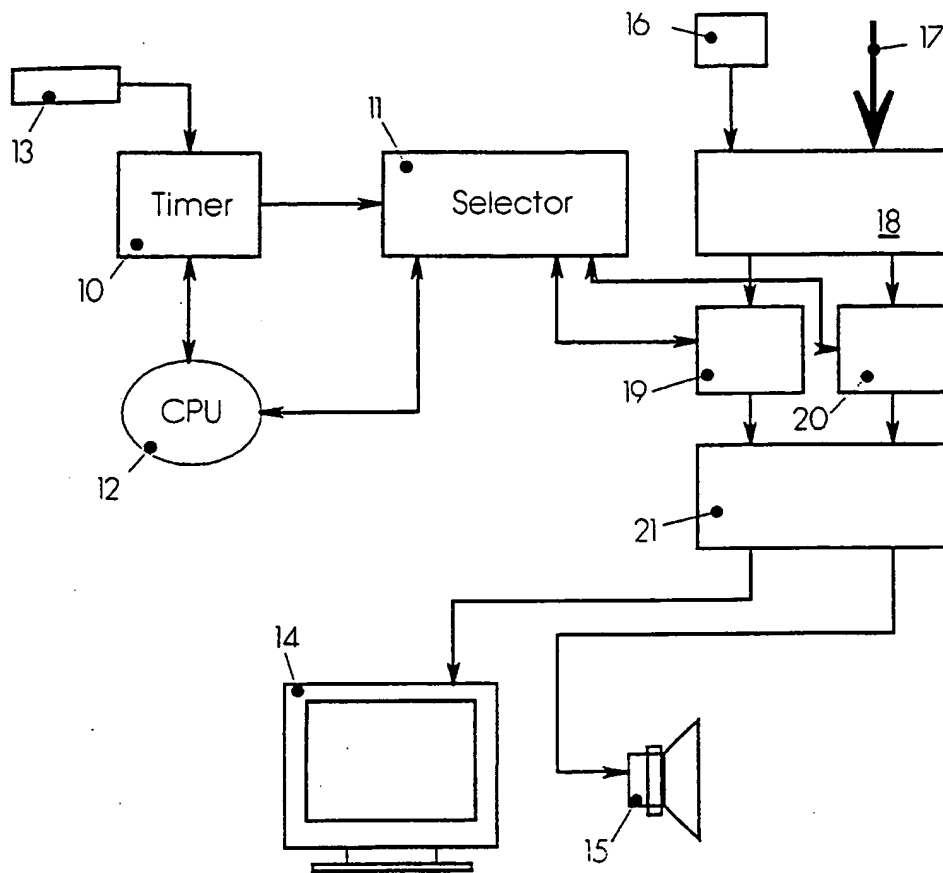


FIG. 1

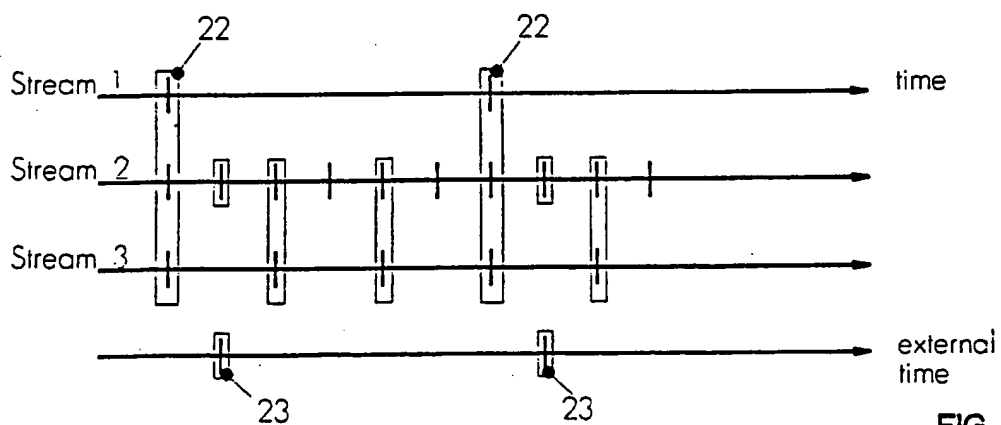


FIG. 2

2/4

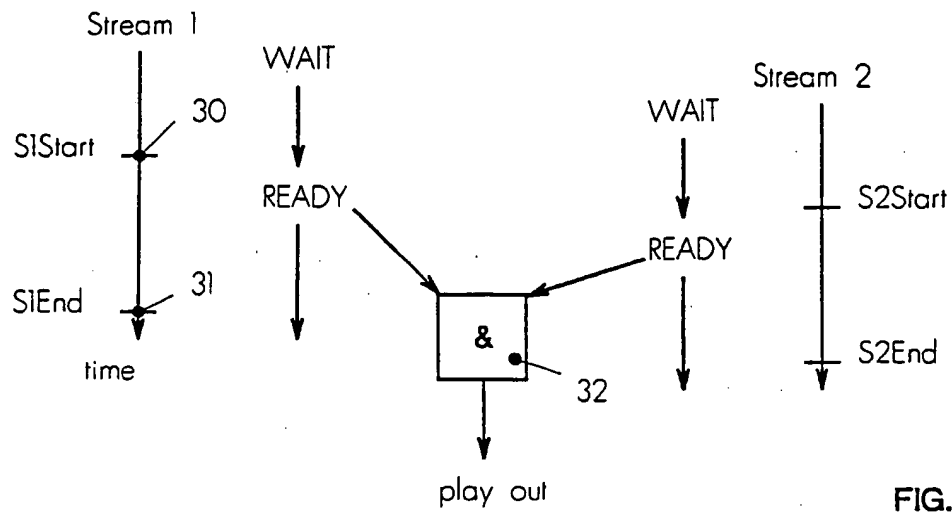


FIG. 3

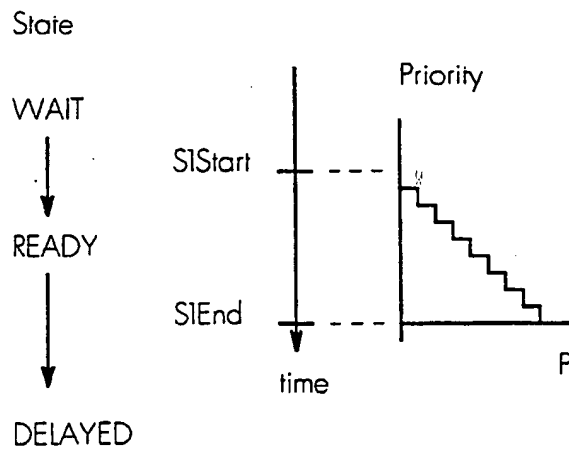
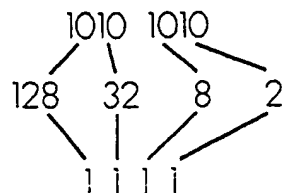


FIG. 4



8-bit Timer

4-bit Priority Field

FIG. 8

3/4

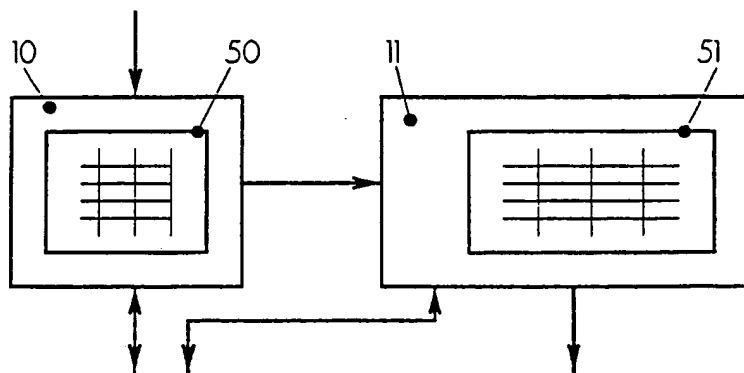


FIG. 5

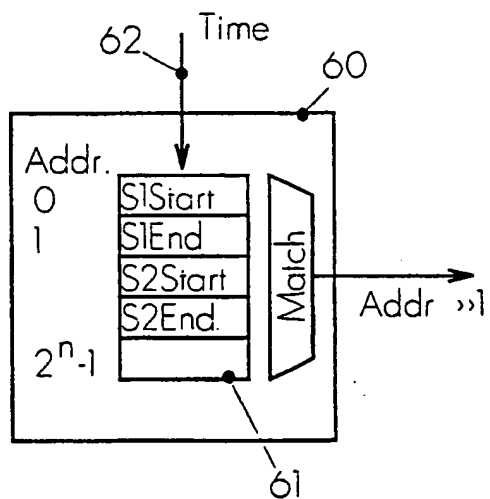


FIG. 6A

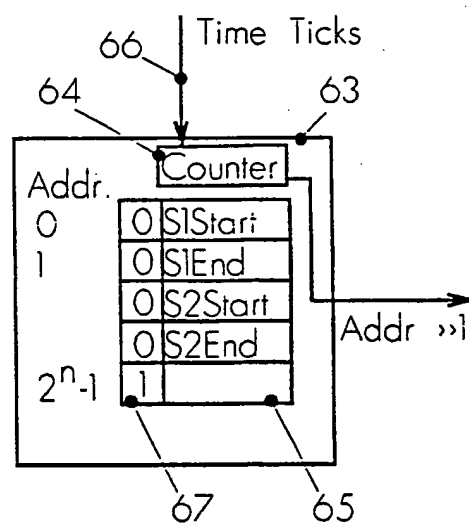


FIG. 6B

4/4

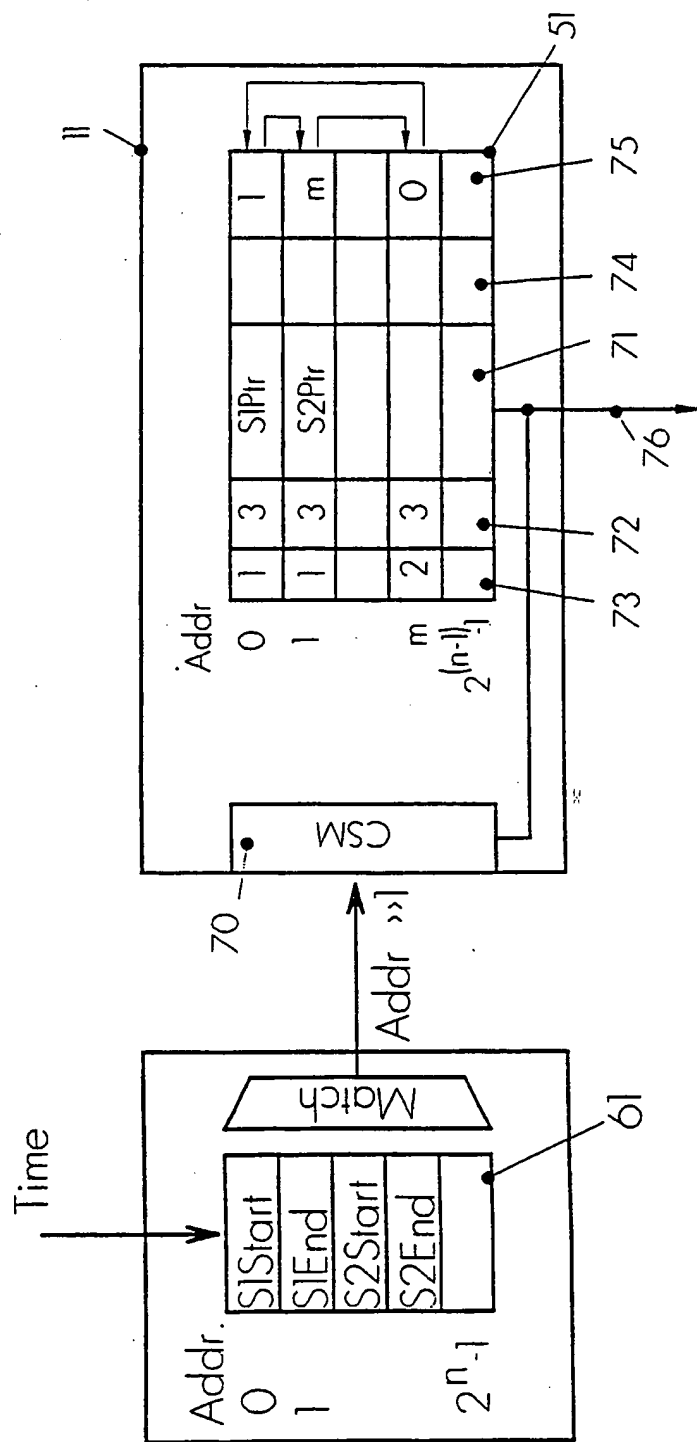


FIG. 7

INTERNATIONAL SEARCH REPORT

International Application No

PCT/EP 94/00833

A. CLASSIFICATION OF SUBJECT MATTER
IPC 6 G06F17/30

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

IPC 6 G06F

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practical, search terms used)

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category *	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	EP,A,0 239 884 (I.B.M. CORPORATION) 7 October 1987 see the whole document ---	1,8,10
A	EP,A,0 550 197 (I.B.M. CORPORATION) 7 July 1993 see the whole document ---	1,8,10
A	COMPUTER COMMUNICATIONS, vol.16, no.3, March 1993, GUILDFORD GB pages 168 - 176 P.V. RANGAN ET AL. 'Techniques for multimedia synchronization in network file systems' see the whole document ---	1,8,10
-/--		

☒ Further documents are listed in the continuation of box C.

☒ Patent family members are listed in annex.

* Special categories of cited documents :

- *A* document defining the general state of the art which is not considered to be of particular relevance
- *E* earlier document but published on or after the international filing date
- *L* document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)
- *O* document referring to an oral disclosure, use, exhibition or other means
- *P* document published prior to the international filing date but later than the priority date claimed

- *T* later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
- *X* document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
- *Y* document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art.
- *&* document member of the same patent family

Date of the actual completion of the international search

10 November 1994

Date of mailing of the international search report

01.12.94

Name and mailing address of the ISA

European Patent Office, P.B. 5818 Patentlaan 2
NL - 2280 HV Rijswijk
Tel. (+31-70) 340-2040, Tx. 31 651 epo nl,
Fax: (+31-70) 340-3016

Authorized officer

Katerbau, R

INTERNATIONAL SEARCH REPORT

Inter. Application No

PCT/EP 94/00833

C.(Continuation) DOCUMENTS CONSIDERED TO BE RELEVANT

Category *	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	<p>COMPUTER, vol.24, no.10, October 1991, LONG BEACH US pages 51 - 57 D.P. ANDERSON ET AL. 'A continuous media I/O server and its synchronization mechanism' see the whole document ---</p>	1,8,10
A	<p>BYTE, vol.18, no.2, February 1993, LONDON, GB pages 225 - 230 A.W. DAVIS ET AL. 'The Mac goes to the movies' see the whole document -----</p>	1,8,10

INTERNATIONAL SEARCH REPORT

Information on patent family members

Intern. Application No

PCT/EP 94/00833

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
EP-A-0239884	07-10-87	US-A- 4893256	09-01-90
		DE-D- 3787553	04-11-93
		DE-T- 3787553	11-05-94
		JP-B- 6054469	20-07-94
		JP-A- 62239233	20-10-87

EP-A-0550197	07-07-93	US-A- 5333299	26-07-94
		CA-A- 2084574	01-07-93
		JP-A- 5265778	15-10-93
